

[First Hit](#) [Fwd Refs](#)[Previous Doc](#)[Next Doc](#)[Go to Doc#](#)**End of Result Set**

Generate Collection

Print

L2: Entry 1 of 1

File: USPT

May 13, 1997

DOCUMENT-IDENTIFIER: US 5630127 A

TITLE: Program storage device and computer program product for managing an event driven management information system with rule-based application structure stored in a relational database

Detailed Description Text (6):

GRMS interfaces with operational systems and management information systems. Operational systems provide GRMS with business event data in the form of transactions or balance updates. Management information systems provide GRMS with reference data such as: product, organization, and economic data. GRMS also interfaces with market data providers to obtain information such as, for example: foreign exchange rates, market prices and counter party ratings. All of this data is collected by the system in order to create the reports needed by management to make decisions.

Detailed Description Text (43):

The lower-level tier consists of these associated programs that derive the values for the data objects. These programs can be simple retrievals of data (e.g., attribute values from CMIM entities) or complex calculations such as an option pricing rule (which in turn may use the object structure to retrieve values for its calculation processing).

Detailed Description Text (54):

The value of the option (option.sub.-- value 208) illustrates an object that is a more complex program that utilizes objects within its logic. These objects, shown as the option's strike price (option-strike-price 216) and the option's duration (option-duration 218), are obtained by accessing the object table in the programs that are associated with those objects.

Detailed Description Text (56):

Similarly, the fourth line of the table 322 shows that the currency exchange rate is obtained from the program IMDCEF10 324. The fifth line of the table 326 shows that the option value is obtained from a program called Black Scholes (Black.sub.-- Scholes.sub.-- Pgm) 312 (a well known complex financial calculation). The sixth line of the table 328 shows that the option duration is obtained by the program IMDCMI22 322. The last line 330 of the table shows that the option strike price is obtained by the program IMDCMI27 334.

Detailed Description Text (82):

The accessing of "Option.sub.-- Value" proceeds along similar lines, although now the program called is not merely accessing the value, but calculating it. It in turn will need to access the object table to have values retrieved for it for the object "Option.sub.-- Duration" and "Option.sub.-- Strike.sub.-- Price". It will use the GRMS processor 108 to achieve this access. As before, the retrieved instances of the objects will be stored in the Object Instance table, in this case in rows 3 414 and 4 416 of FIG. 4.

Detailed Description Text (139):

The Market Information Handling component creates a standard interface for accessing market information in support of the calculation of exposure and risk values as well as the derivation of volatilities associated with market entities (e.g., equity prices, interest rates, currency exchange rates).

Detailed Description Text (195):

Data stored in the CMIM defined CMIS database. This data is required by multiple CMIS business applications. It includes bank organizational data, customer data, product attributes, account details, performance measurements, price data.

Detailed Description Text (246):

Data stored in the CMIM defined CMIS database. This data is required by multiple CMIS business applications. It includes bank organizational data, customer data, product attributes, account details, performance measurements, price data.

Detailed Description Text (283):

Data stored in the CMIM defined MIS database. This data is required by multiple CMIS business applications. It includes bank organizational data, customer data, product attributes, account details, performance measurements, price data.

[Previous Doc](#)

[Next Doc](#)

[Go to Doc#](#)

[First Hit](#) [Fwd Refs](#) [Previous Doc](#) [Next Doc](#) [Go to Doc#](#)**End of Result Set**☐ [Generate Collection](#) [Print](#)

L3: Entry 1 of 1

File: USPT

May 13, 1997

DOCUMENT-IDENTIFIER: US 5630127 A

TITLE: Program storage device and computer program product for managing an event driven management information system with rule-based application structure stored in a relational database

Drawing Description Text (6):

FIG. 4 is an object instance table.

Detailed Description Text (6):

GRMS interfaces with operational systems and management information systems. Operational systems provide GRMS with business event data in the form of transactions or balance updates. Management information systems provide GRMS with reference data such as: product, organization, and economic data. GRMS also interfaces with market data providers to obtain information such as, for example: foreign exchange rates, market prices and counter party ratings. All of this data is collected by the system in order to create the reports needed by management to make decisions.

Detailed Description Text (43):

The lower-level tier consists of these associated programs that derive the values for the data objects. These programs can be simple retrievals of data (e.g., attribute values from CMIM entities) or complex calculations such as an option pricing rule (which in turn may use the object structure to retrieve values for its calculation processing).

Detailed Description Text (50):

GRMS adds a table driven structure in front of these accessing programs so that the attribute representations can be referenced by name; these names can be thought of as variables that would appear in mathematical formulas. An example of this table is shown in FIG. 3. The names or "objects" are shown in the columns "OBJECT" 302, "OBJECT1" 304 and "OBJECT2" 308. These names or "objects" stand for a multitude of particular instances of the data, any of which can be retrieved by specifying the identifiers of the entities listed above which would focus the access on a particular representation value.

Detailed Description Text (54):

The value of the option (option.sub.-- value 208) illustrates an object that is a more complex program that utilizes objects within its logic. These objects, shown as the option's strike price (option-strike-price 216) and the option's duration (option-duration 218), are obtained by accessing the object table in the programs that are associated with those objects.

Detailed Description Text (56):

Similarly, the fourth line of the table 322 shows that the currency exchange rate is obtained from the program IMDCEF10 324. The fifth line of the table 326 shows that the option value is obtained from a program called Black Scholes (Black.sub.-- Scholes.sub.-- Pgm) 312 (a well known complex financial calculation). The sixth line of the table 328 shows that the option duration is obtained by the program

IMDCMI22 322. The last line 330 of the table shows that the option strike price is obtained by the program IMDCMI27 334.

Detailed Description Text (80):

The processor then would check for a prior retrieval of this value in its Object Instance table (for example, 400 of FIG. 4) associated with this report. Since this is the first access, the table would be empty, and the program would complete the processing of the retrieval from the actual database. Once the value is retrieved, the last action before returning would be the creation of an entry in the GRMS Object Instance Table for the value. This entry is shown as the first row 410 of the table 400 in FIG. 4. A similar processing would be done for the "Currency.sub.-- Exch.sub.-- Rate", which would result in the second entry in the table 400 of FIG. 4. Each time a rule is executed, an object instance table containing the primitive value used by the rule is created.

Detailed Description Text (82):

The accessing of "Option.sub.-- Value" proceeds along similar lines, although now the program called is not merely accessing the value, but calculating it. It in turn will need to access the object table to have values retrieved for it for the object "Option.sub.-- Duration" and "Option.sub.-- Strike.sub.-- Price". It will use the GRMS processor 108 to achieve this access. As before, the retrieved instances of the objects will be stored in the Object Instance table, in this case in rows 3 414 and 4 416 of FIG. 4.

Detailed Description Text (83):

Once the program returns the value for the "Option.sub.-- Value", the GRMS Process 108 now has the values for both objects in the initial expression and can calculate the actual value for the Option.sub.-- Exposure as the product of the two values. At this point the value is returned and placed into the report that is shown to the customer. As part of this report process, the Object Instance table 400 created in this processing and shown in FIG. 4 is returned to the Business Professional's workstation 118 for reference purposes.

Detailed Description Text (84):

An additional feature of the GRMS architecture is the placement of the GRMS processor on the Business Professional's workstation 118 along with the Object Table 300, and the programs defined in the object table 300. Since the object instance table 400 is also present, the Business Professional can change values in the Object Instance table (via GRMS screens and functions) and reprocess the report on the workstation. All object accesses will be satisfied by the Object Instance table function and therefore, the CMIM database 224 is not needed for this "What if" analysis reporting.

Detailed Description Text (87):

This final program would follow the standard accessing protocol described above. In execution, this translated program is executed in a similar manner to the interpretive processing of the GRMS Processor, but would be faster. An object instance table would be created per each report request as objects are accessed and values retrieved.

Detailed Description Text (100):

For example, in FIG. 14 the data block labelled "Obj to Mod" 1410 is the table referred to in the description as the Object Table 300 ("Module" is a technical term for program). The data block labelled "Object Instance" 1408 represents the Object Instance table 400 in the description. "Data Obj Def" 1402 refers to a definition of the object to be retrieved in terms of its data type (e.g., Integer or Character), length, significance if a decimal point is involved, etc. The additional data in the diagram describes the messaging connectivity of the systems.

Detailed Description Text (139):

The Market Information Handling component creates a standard interface for accessing market information in support of the calculation of exposure and risk values as well as the derivation of volatilities associated with market entities (e.g., equity prices, interest rates, currency exchange rates).

Detailed Description Text (147):

When a query is run on the host, all of the data used in running that query is stored in an "Object Instance Table." This table is sent to the business professional at his/her workstation and contains a diverse set of information: product names, customer identifiers, exposure values, risk factors-everything that was calculated as part of the query or found in any database table.

Detailed Description Text (150):

To support additional analysis, the object instance table can be imported into spreadsheet, graphics, or other analytical tools.

Detailed Description Text (151):

GRMS uses the CMIS database which is a relational database established and shared by other CMIS Applications (i.e., Customer Information, Product Management and Balance Sheet and Liquidity Management). The Integrated Database Component unites the various databases that comprise the GRMS system, as illustrated in FIG. 7. This figure shows configuration with multiple primary nodes 704 and 706. The Enterprise Database 702 would be the one for the entire bank while the other primary nodes' databases would reflect data for subunits of the enterprise (e.g., a domestic unit, foreign unit or foreign regional). Also illustrated is the connection 708 between a primary node (in this case the Enterprise node 702) and a workstation where the database (primary node) and the data tables 710 (secondary workstation node) are related by the Integrated Database components in each. Finally, the figure illustrates the connection of the object instance table 712 that, as a result of a query or report, is developed on the primary node and transferred to the secondary mode for use in query and report interpretation and risk analysis.

Detailed Description Text (195):

Data stored in the CMIM defined CMIS database. This data is required by multiple CMIS business applications. It includes bank organizational data, customer data, product attributes, account details, performance measurements, price data.

Detailed Description Text (217):

Exposure report events triggered by an exposure calculation, a user query request and/or calendar event that meets the rule instance to automatically execute the exposure reporting process.

Detailed Description Text (246):

Data stored in the CMIM defined CMIS database. This data is required by multiple CMIS business applications. It includes bank organizational data, customer data, product attributes, account details, performance measurements, price data.

Detailed Description Text (256):

Rules instances specified by the 'Risk Perspective' to be executed in computing Probable Loss.

Detailed Description Text (268):

Selected risk details describing instances of probable loss calculations.

Detailed Description Text (283):

Data stored in the CMIM defined MIS database. This data is required by multiple CMIS business applications. It includes bank organizational data, customer data, product attributes, account details, performance measurements, price data.

Detailed Description Text (289):

Updated rule instances created and/or modified and tested by the business professional at the workstation. This data may also be received via imported data from other GRMS installations.

Detailed Description Text (297):

Requests describing the requirements for instances of data stored within the CMIS database.

Detailed Description Text (309):

Selected risk details describing instances of probable loss calculations.

Detailed Description Text (318):

This process receives the results of queries/reports from the primary node, presents them to users, and stores the object instance data to the local database for Risk Analysis. Other activities can also be received from the host, including messages from other users, and messages from behavior rule (e.g., limit) processing.

Detailed Description Text (320):

Risk analysis allows business users to execute queries/reports against instance data received from the primary node after a query/report execution. The user can change object instance values to perform "what if" analysis.

Detailed Description Text (326):

The Object Instance entity on the primary node stores all the values retrieved and calculated during the execution of queries/reports. These values are sent, through activity management, to the secondary nodes where the queries/reports were requested, and stored in this entity. Users can use these object instance values in performing risk and other "what if" analysis by changing the object instance values and running local queries/reports.

Detailed Description Text (334):

Updated rule instances created and/or modified and tested by the business professional at the workstation. This data may also be received via imported data from other FILMS installations.

Detailed Description Text (342):

Data sent from the primary node to a secondary node including report output, object instance data for risk analysis, and message data from their users or from behavior rule (limit) processing.

Detailed Description Text (350):

Request for object values from the object instance datastore.

Detailed Description Text (364):

The GRMS Data Classes subject areas represent Exposure, Risk, GRMS Factors, Object Instance data and their relationships to those CMIM entities whose IDs are used within the keys. GRMS Data Classes contain the data created and used by the GRMS unique risk factors (e.g., credit risk factors) and specific instances of data to be analyzed within GRMS. The subject area contains the following entities:

Detailed Description Text (365):

The System Parameter, Code, Business Calendar, Security Authorization entities area used to establish system parameters, look up tables, calendars, and to control application access. The Perspective and User Defined Facility entities are created by the GRMS user to define the 'scenario' used to calculate risk and view the results by a facility other than those defined within CMIM. The Exposure DB, Risk DB, Object Instance and GRMS Factor contain the core data of the GRMS application data. Each of these entities is keyed by the ID of CMIM entities together with risk

types and timestamp information. This key structure provides the GRMS application with historic data describing the institution's exposure and risk together with the factors and specific object data used to calculate that information. The remaining entities are used within the CMIM product to describe the institution's organization, customers, portfolio and etc.

Detailed Description Text (375):

Whenever an object routine is executed, it places the returned value in the Object Instance entity. After executing a query/report, all the Object Instance values resulting from that query/report can be downloaded to the workstation for analysis. This analysis allows the query/report can be re-run on the workstation, whose object routines will find their values in the Object Instance entity. By changing these values and re-running queries/reports, the business professional can perform "what if" analysis.

Detailed Description Text (430):

The following is a description of Entity Type: Instance 1408:

Detailed Description Text (435):

This is a set of object specific information which makes this instance of an object different from other instances. If this object instance is uniquely identified by the 10 part CMIM key, this field will be null.

Detailed Description Text (437):

The value of this object instance. This will be a number for calculations and numeric lookups, or a character string for text lookups.

Detailed Description Text (439):

Unique identifier for a GRMS transaction. A single GRMS transaction can generate many object instances. See GRMS Queue entity.

Detailed Description Text (441):

This relates an object and access type with the value in the object instance entity.

Detailed Description Text (442):

--Relationship Type: PRODUCT.Identifies.Obj Instance

Detailed Description Text (443):

Relates object instances to PRODUCT portion of CMIM key.

Detailed Description Text (444):

--Relationship Type: MANAGEMENT UNIT.Identifies.Obj Instance

Detailed Description Text (445):

Relates object instances to MANAGEMENT UNIT portion of CMIM key.

Detailed Description Text (446):

--Relationship Type: MARKET SEGMENT.Identifies.Obj Instance

Detailed Description Text (447):

Relates object instances to MARKET SEGMENT portion of CMIM key.

Detailed Description Text (448):

--Relationship Type: LEGAL PARTICIPANT.Identifies.Obj Instance

Detailed Description Text (449):

Relates object instances to LEGAL PARTICIPANT portion of CMIM key.

Detailed Description Text (450):

--Relationship Type: CUSTOMER.Identifies.Obj Instance

Detailed Description Text (451):

Relates object instance to CUSTOMER portion of CMIM key.

Detailed Description Text (452):

--Relationship Type: CUSTOMER ACCOUNT.Identifies.Obj Instance

Detailed Description Text (453):

Relates object instances to CUSTOMER ACCOUNT portion of CMIM key.

Detailed Description Text (454):

--Relationship Type: PORTFOLIO.Identifies.Obj Instance

Detailed Description Text (455):

Relates object instances to PORTFOLIO portion of CMIM key.

Detailed Description Text (456):

--Relationship Type: USER DEFINED FACILITY.Identifies.Obj Instance

Detailed Description Text (457):

Relates object instances to USER DEFINED portion of CMIM key.

Detailed Description Text (458):

--Relationship Type: FIRM ACCOUNT.Identifies.Obj Instance

Detailed Description Text (459):

Relates object instances with the FIRM ACCOUNT portion of the CMIM key.

Detailed Description Text (460):

--Relationship Type: GEOGRAPHIC REGION.Identifies.Obj Instance

Detailed Description Text (461):

Relates object instances to the GEOGRAPHIC REGION portion of the CMIM key.

Detailed Description Text (469):

--Relationship Type: Obj Instance. Generated By.Obj to Mod

Detailed Description Text (470):

This relates an object and access type with the value in the object instance entity.

Detailed Description Paragraph Table (3):

	Data Class Entities Entity System
	System Parameter GRMS Code GRMS Business
Calendar GRMS Security Group GRMS Security Authorization GRMS Perspective GRMS User	
Defined Facility GRMS Exposure DB GRMS Risk DB GRMS Object <u>Instance</u> GRMS GRMS	
Factor GRMS EMPLOYEE CMIS MANAGEMENT UNIT CMIS PRODUCT CMIS MARKET SEGMENT CMIS	
GEOGRAPHIC REGION CMIS MKT SEG CUSTOMER CMIS LEGAL PARTICIPANT CMIS CUSTOMER CMIS	
CUSTOMER ACCOUNT CMIS CUST ACCT PORT CMIS PORTFOLIO CMIS FIRM ACCOUNT CMIS ACCOUNT	
CATALOG CMIS	

Detailed Description Paragraph Table (7):

	Attribute Types: key- ID Req Tran Num
(LONGINT) Req Obj to Mod.Object ID *1 Req PRODUCT.key *2 Opt MANAGEMENT UNIT.key *3	
Opt MARKET SEGMENT.key *4 Opt GEOGRAPHIC REGION.key *5 Opt LEGAL PARTICIPANT.key *6	
Opt CUSTOMER ACCOUNT.key *7 Opt CUSTOMER.key *8 Opt PORTFOLIO.key *9 Opt FIRM	
ACCOUNT.key *10 Opt USER DEFINED FACILITY.key *11 Opt <u>Instance</u> Context	
(DESCRIPTION) Opt <u>Instance</u> Value (LONGDESC) Req Relationship Types Generated By.Obj	
to Mod *1 1-1 0-M Identified By.PRODUCT *2 0-1 0-M Identified By.MANAGEMENT UNIT *3	

0-1 0-M Identified By.MARKET SEGMENT *4 0-1 0-M Identified By.GEOGRAPHIC REGION *5
0-1 0-M Identified By.LEGAL PARTICIPANT *6 0-1 0-M Identified By.CUSTOMER ACCOUNT
*7 0-1 0-M Identified By.CUSTOMER *8 0-1 0-M Identified By.PORTFOLIO *9 0-1 0-M
Identified By.FIRM ACCOUNT *10 0-1 0-M Identified By.USER DEFINED FACILITY *11 0-1
0-M

Detailed Description Paragraph Table (8):

It relates . . . To modules which are dynamically executed . . . Process objects in response to messages. Process objects use data objects and calculation objects to perform data access and computations. (Examples: Exposure and risk processes, report and query processes). Data objects and access during the execution of other types (create, read, update, objects or by the base applica- delete) tion. These objects can access data anywhere on the institu- tion's system (CMIS, GRMS, or other systems/databases). Examples: Exposure Value/read, Risk/create, Product Name/read, Factor/create, Code/delete, Sys Parm/read.) Calculation objects during the execution of other objects or by the base application. The access type for calculation objects is always "read." (Examples: Black Shoals calculation, exposure and risk calculations, multiply risk times risk factor.) Limit objects during the execution of behavior rules processing. These objects will aggregate data in the risk and exposure entities for comparison with limit values. The access type for limit objects is always "read." (Examples: Aggregate exposure for Tokyo management unit, Aggregate risk for all loans.) Attribute Types: key- ID Req Object.Object ID * Req Access Type (SHORTID) Req Relationship Types Identified By.Object * 1-1 0-M Generates.Obj Instance 0-M 1-1 Identifies.Module * 1-1 0-M

[Previous Doc](#)

[Next Doc](#)

[Go to Doc#](#)